

**19. März 2013** in Ehningen



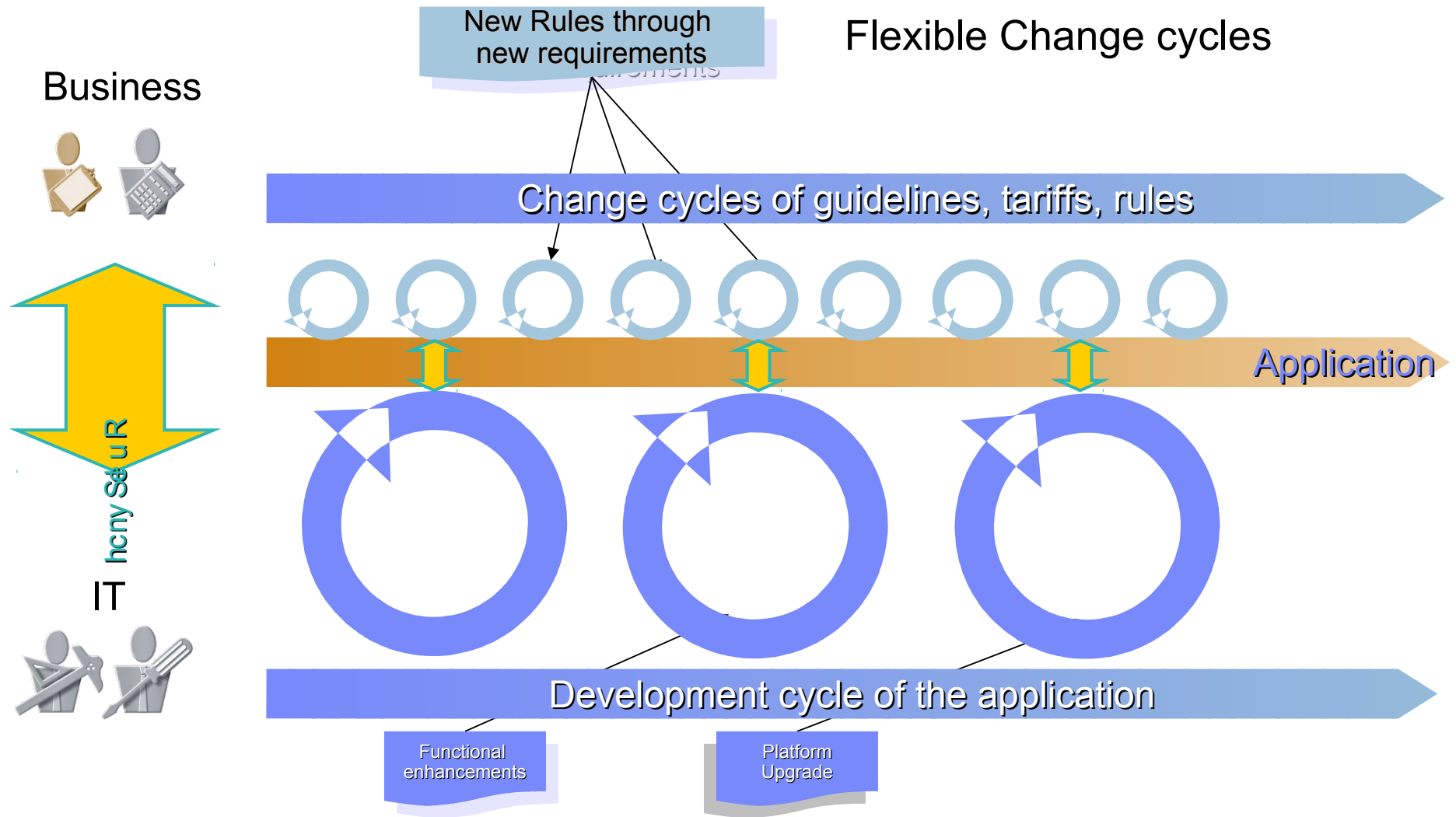
# **Modernisierung von Hostanwendungen mit Java und Geschäftsregeln**

**Joerg-Ulrich Veser**

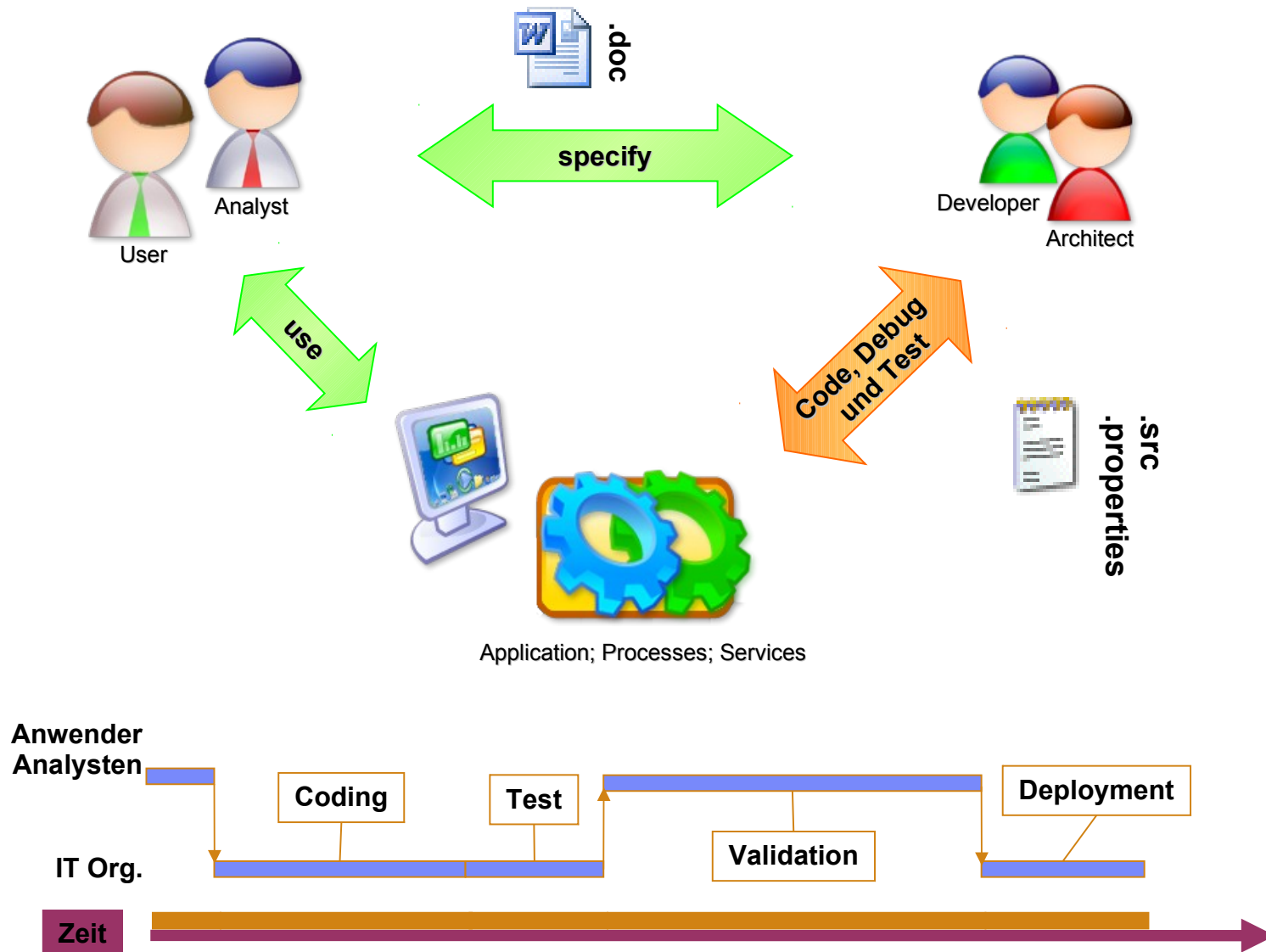
| Client Technical Professional

WebSphere on System z | IBM Germany

# Business and IT-Alignment



# A development cycle without rules



## Business rule mining

- If the monthly repayment for the loan is lower than 33% of the monthly income, it is a customer with low risk.
- If the initial payment is more than 30% a borrower is low risk gets a discount of 1%.

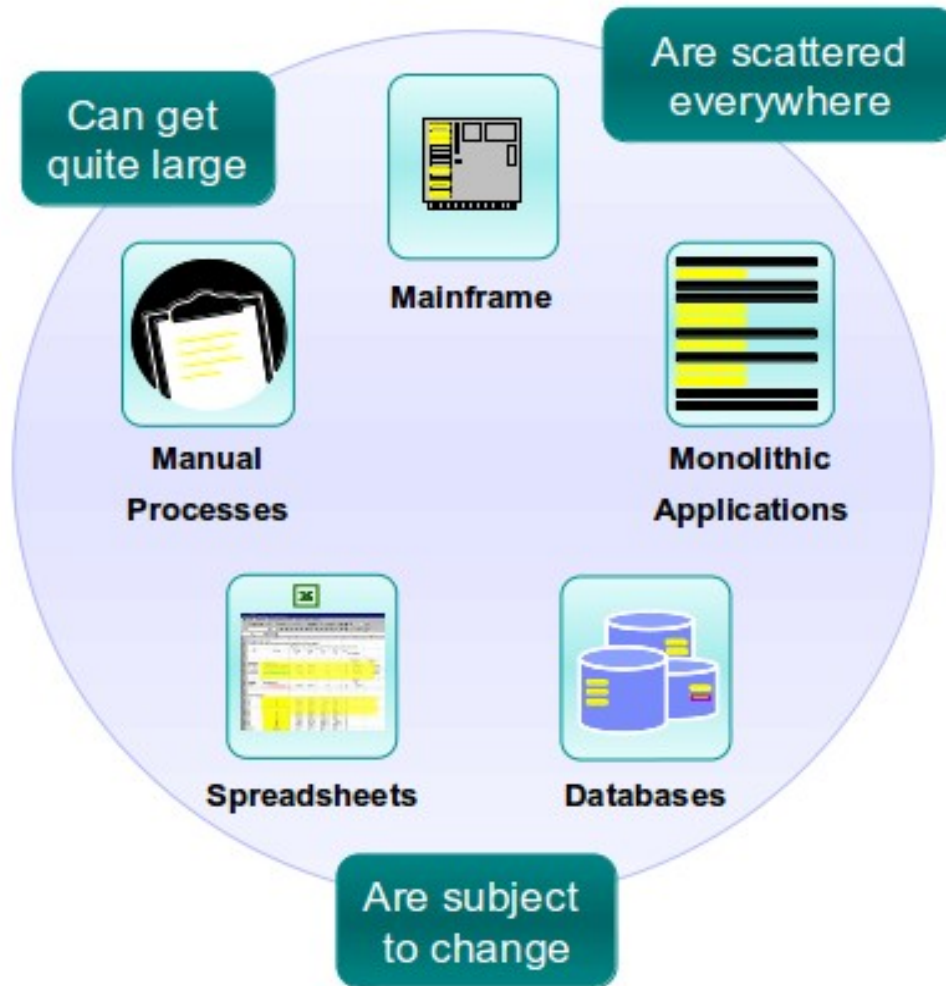
```
public class Borrower {  
    private float outcomes;  
    private float incomes;  
    private float SpecialRate;  
    ...  
    public boolean IsLowRisk () {  
        if (outcomes < (incomes * 0.33)) {  
            return true;  
        } else return false;  
    }  
}
```



Developer

```
public class LoanRequest {  
    private Borrower borrower;  
    private float loanAmount;  
    private float downPayment;  
    ...  
    public void QualifyForLowerRate() {  
        if (borrower.LowRisk()) {  
            if (downPayment > (loanAmount * 0.30))  
                borrower.setSpecialRate (-0.01);  
        }  
    }  
}
```

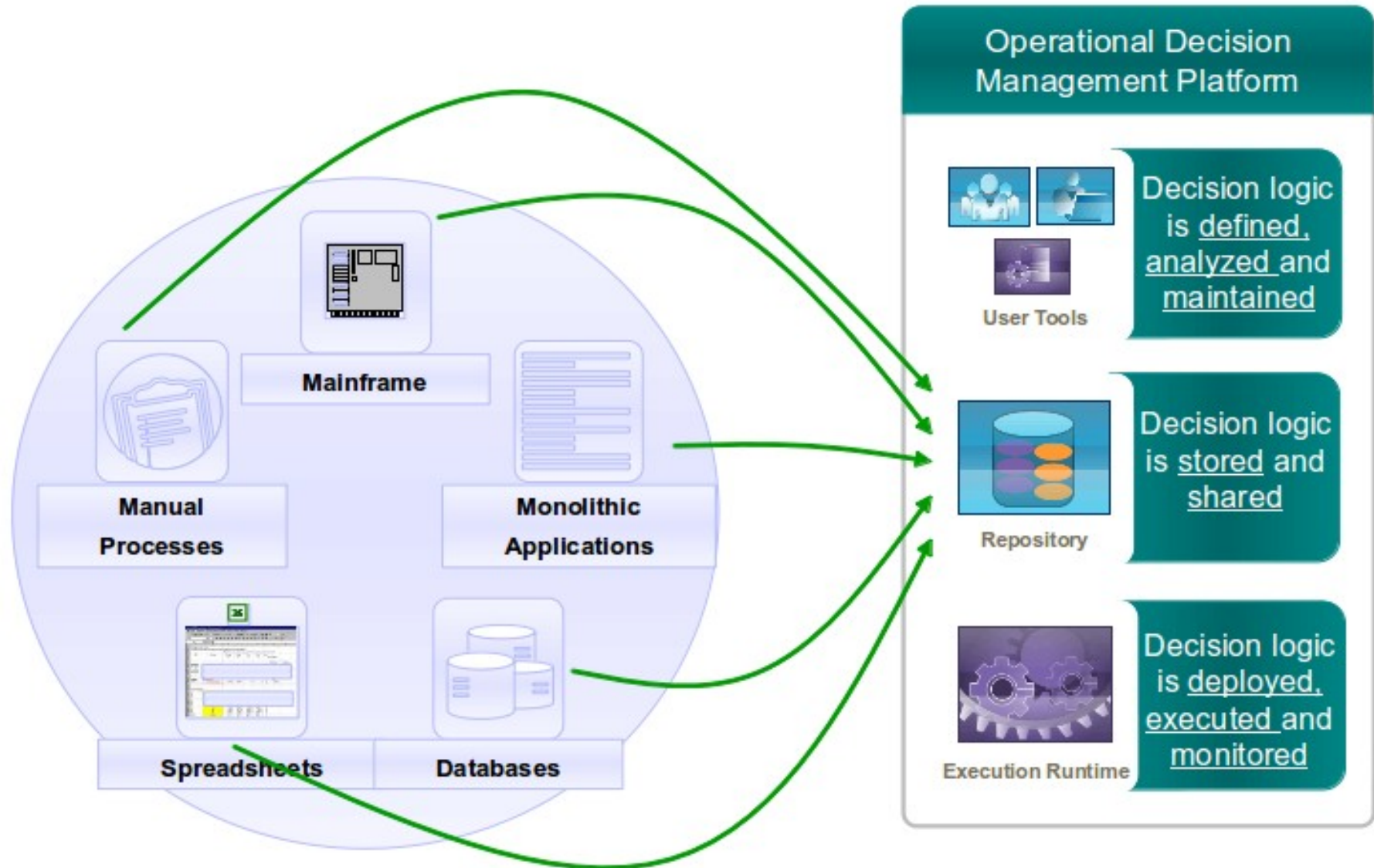
# Operational decisions in organizations



## Challenges for a Change Request

- Changes are costly, resource & time-intensive
  - Hidden in code
  - Most changes have to be programmed – costly
- Lack of consistency
  - No central management
  - No reuse of decision logic
- Gap between business analysts & IT administrators
  - Knowledge fades over time
- Lack of audit ability
- No easy way to test/simulate changes

# Operational Decision Management Approach





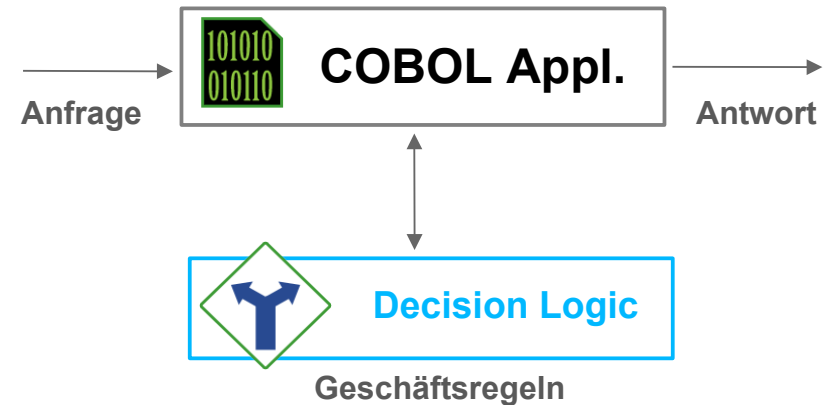
## Extract Decision Logic out of existing applications

Stand heute



Die Entscheidungslogik ist Bestandteil der Anwendung

Nach Modernisierung

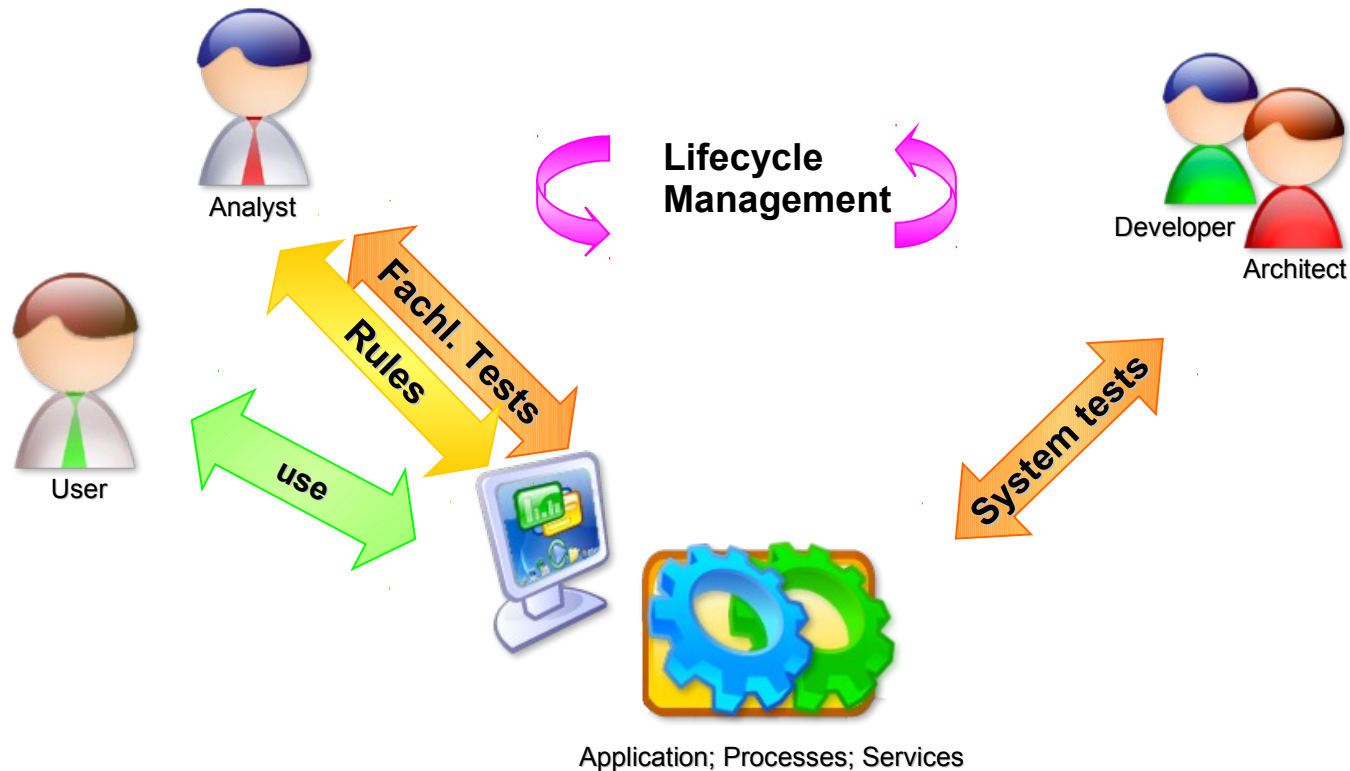


Die Entscheidungslogik wurde in Form von Geschäftsregeln ausgelagert.

### Vorteile für ein separates Management von Geschäftsregeln:

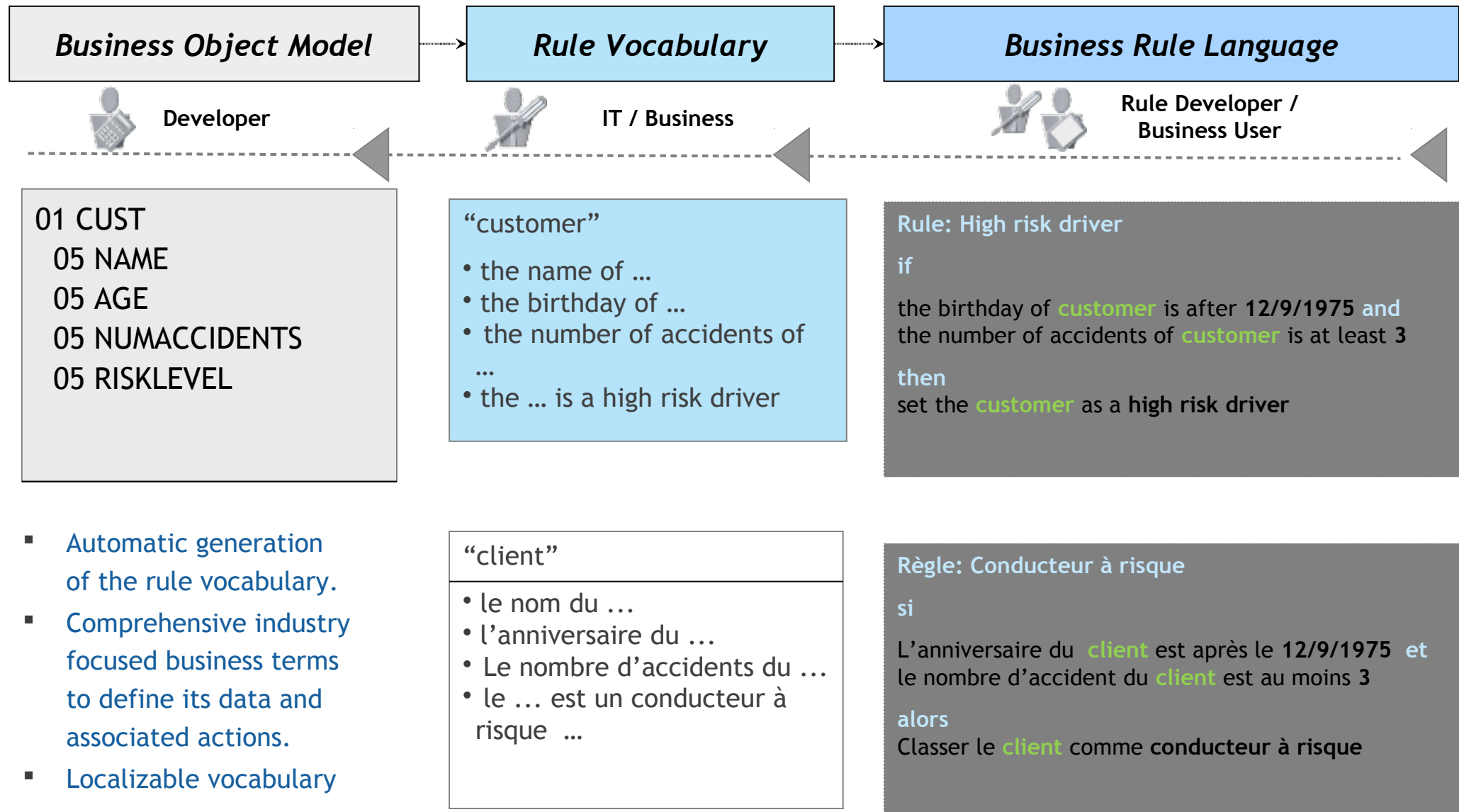
- Geschäftsregeln in natürlicher Sprache können einfacher verstanden werden
- Externalisierte Regeln können einfacher geändert werden
- Zentralisierte Regeln ermöglichen eine **Wiederverwendung und Konsistenz**

## A development cycle rules





## Data Model - Verbalization



- Automatic generation of the rule vocabulary.
- Comprehensive industry focused business terms to define its data and associated actions.
- Localizable vocabulary

# Rules vs. Events

## Business Events

Primarily implements a time-based pattern detection model – correlating events as data is in motion

Main purpose is to determine what of interest is transpiring and coordinate one or more responses by other systems or generate alerts to people

If more than 2 **ATM** withdrawals from same **account** are done **in the same day** and the 2 **ATM** transactions are from 2 foreign countries  
Then **Investigate possible fraud**

## Business Rules

Primarily implements a decision model – given a snapshot view of data, determines best course of action at a specific point in a process or application

Main purpose is to automate a decision based on a combination of factors (business policies, regs, best practices)

If the **customer** is not **primary card holder** and **age** is less than 21 then  
Freeze **account** and notify **primary card holder**

## Event Pattern Detection

### CICS Events

07:00	08/09/2012	ATM	0123	WITHDRAW	\$ 400.00	CANNES
07:09	08/09/2102	WEB	2405	TRANSFER	\$1000.00	USA
08:09	08/09/2102	TEL	2948	DEPOSIT	\$ 269.00	USA
12:00	08/09/2102	ATM	0123	WITHDRAW	\$ 400.00	NICE
12:01	08/09/2102	WEB	9485	BILLPAY	\$ 294.00	USA
13:04	08/09/2102	TEL	8204	WITHDRAW	\$2000.00	USA
14:10	08/09/2102	ATM	0123	WITHDRAW	\$ 500.00	MONACO

Event  
Pattern  
Match

### Business Event Pattern

If more than 2 **ATM** withdrawals  
from same **account**  
are done in the **same day**  
and the 2 **ATM** transactions are from 2  
foreign countries  
Then **Investigate possible fraud**

True

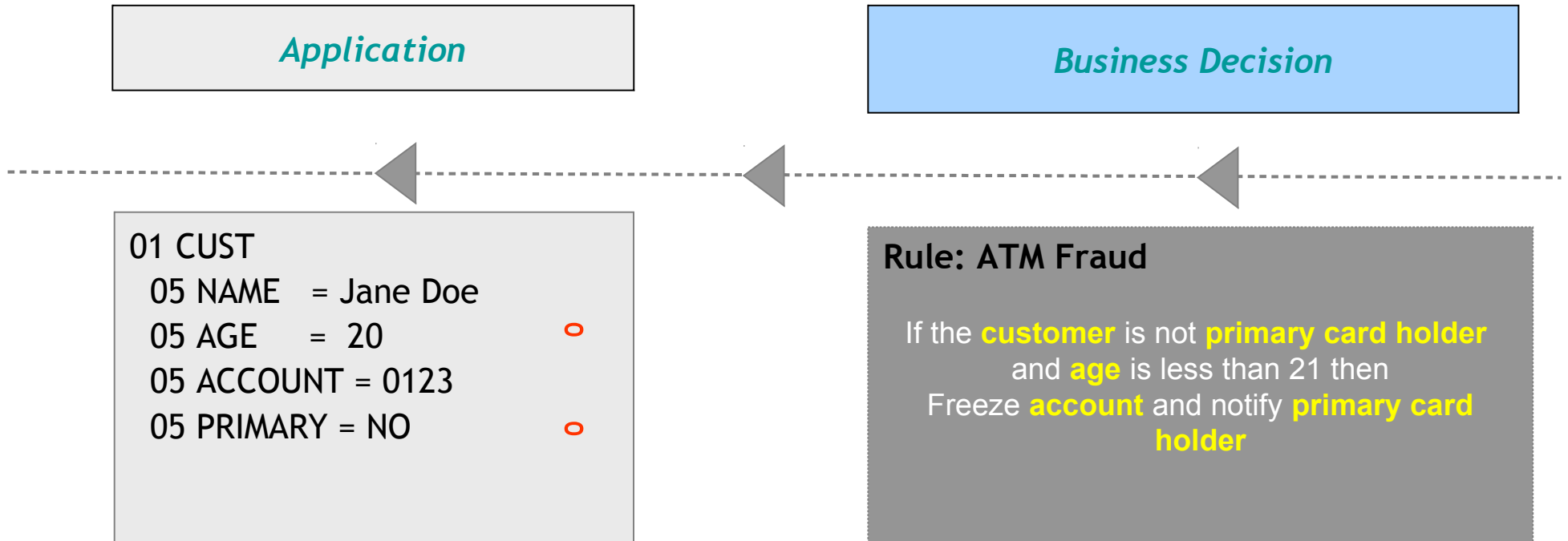
True

True

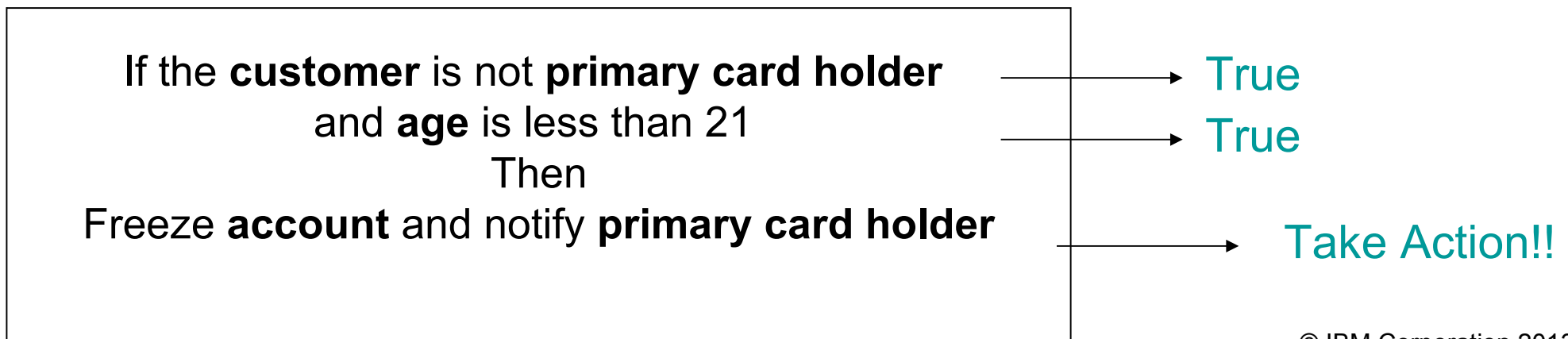
True

Pass to Business Decision

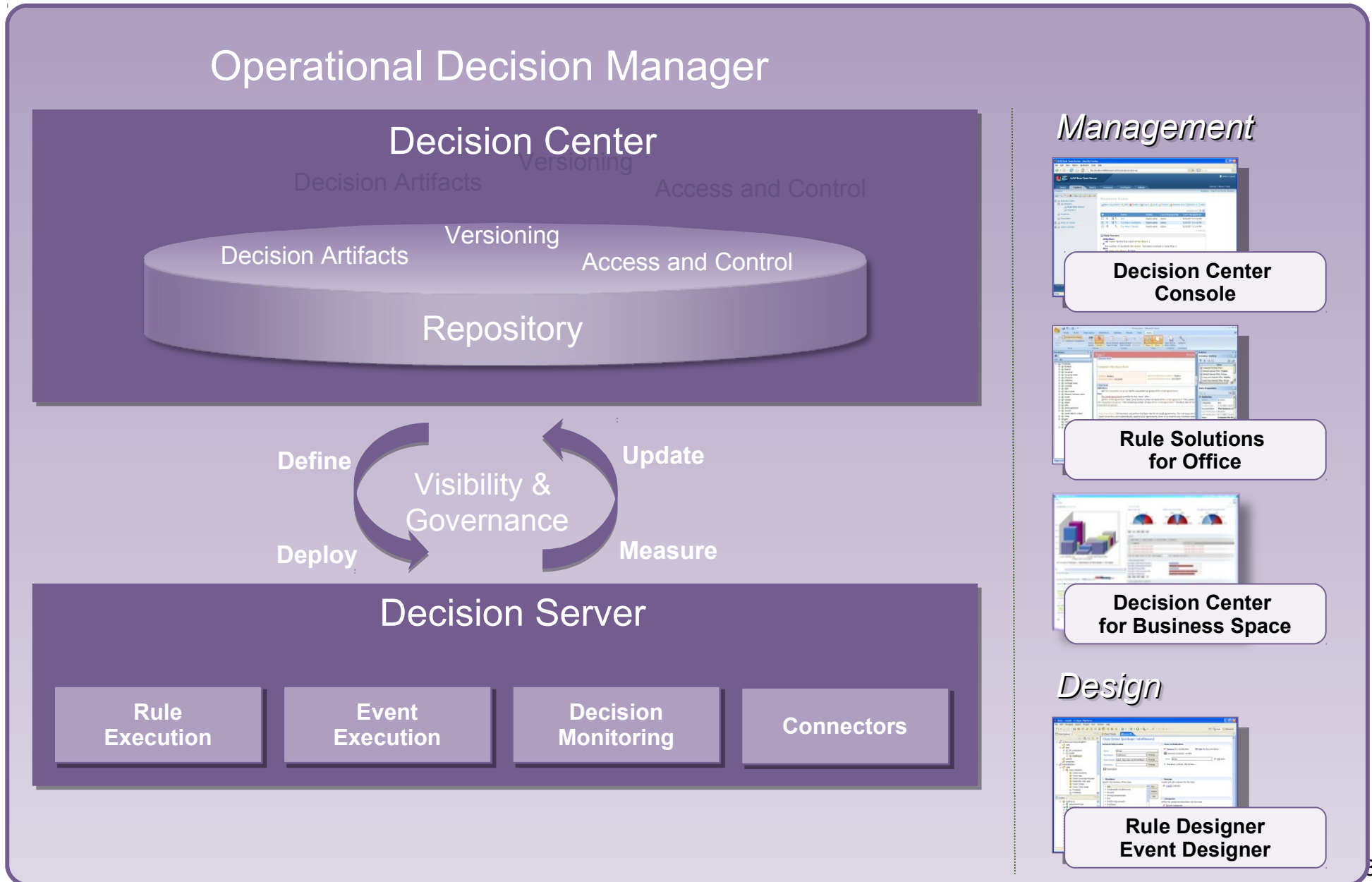
## Event Driven Decision to Act



### Business Rule

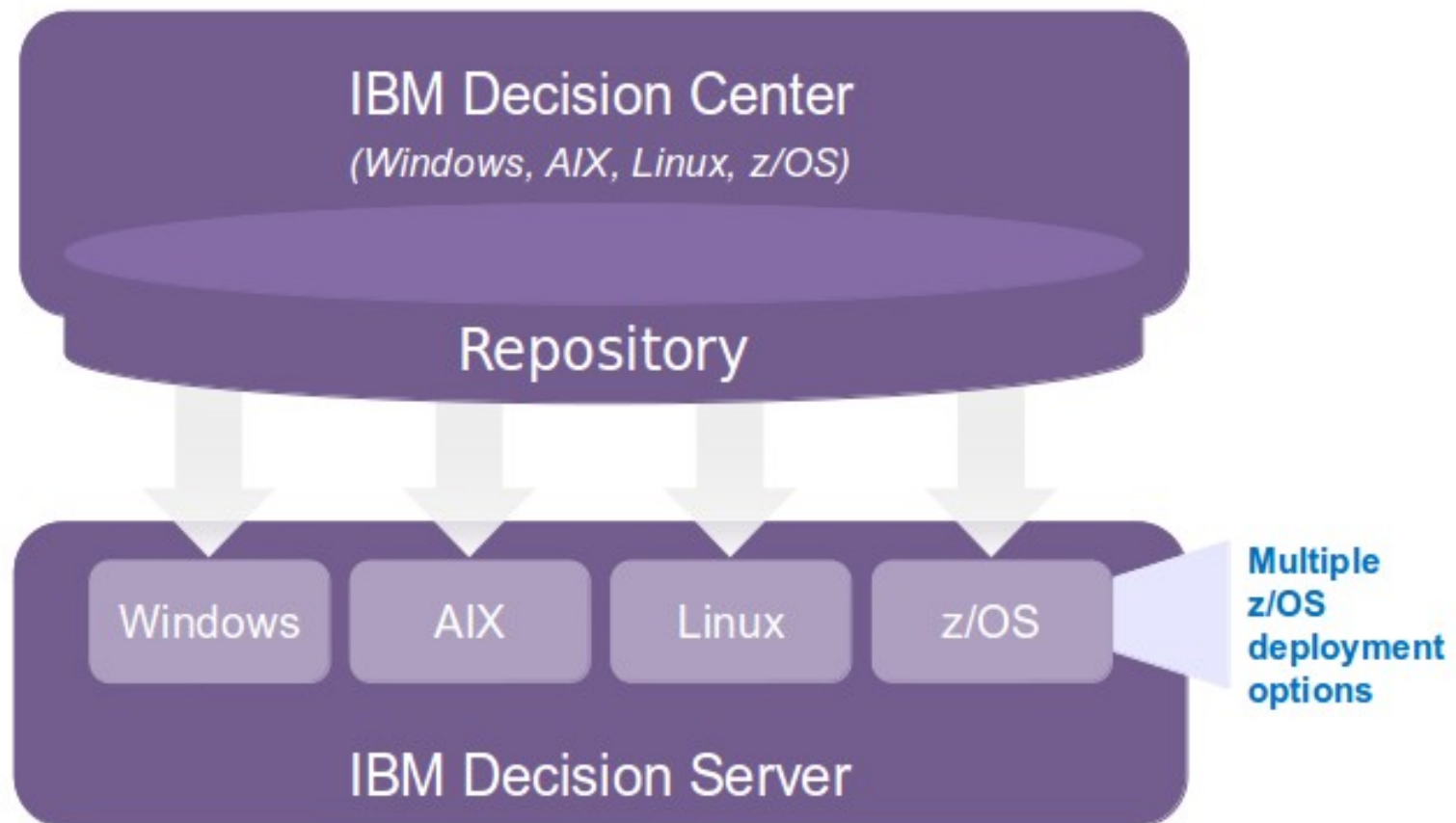


# IBM Operational Decision Manager - Components



# IBM Operational Decision Manager – Runtime Support

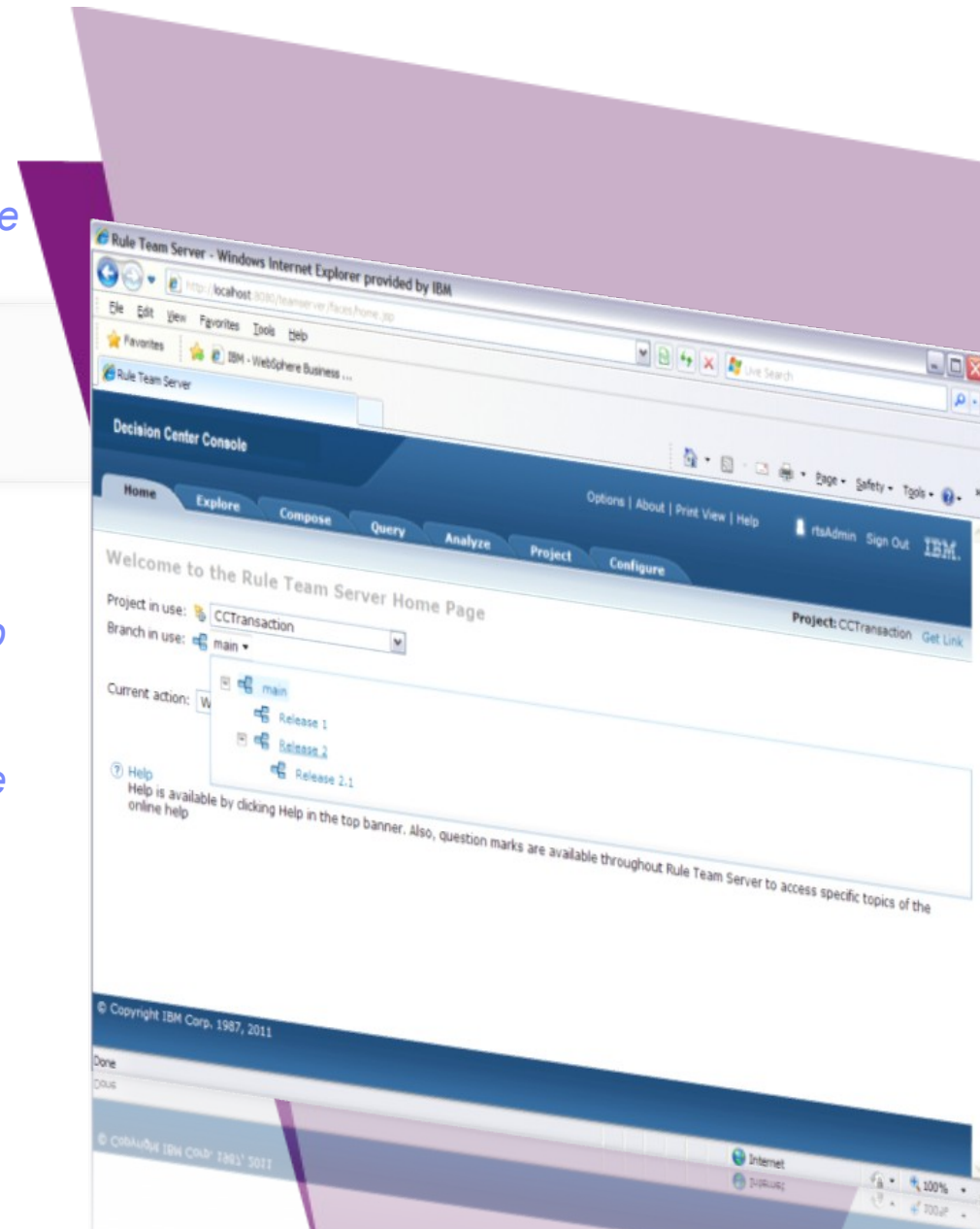
Leverage a wide range of platforms to meet the varying needs of enterprise architectures



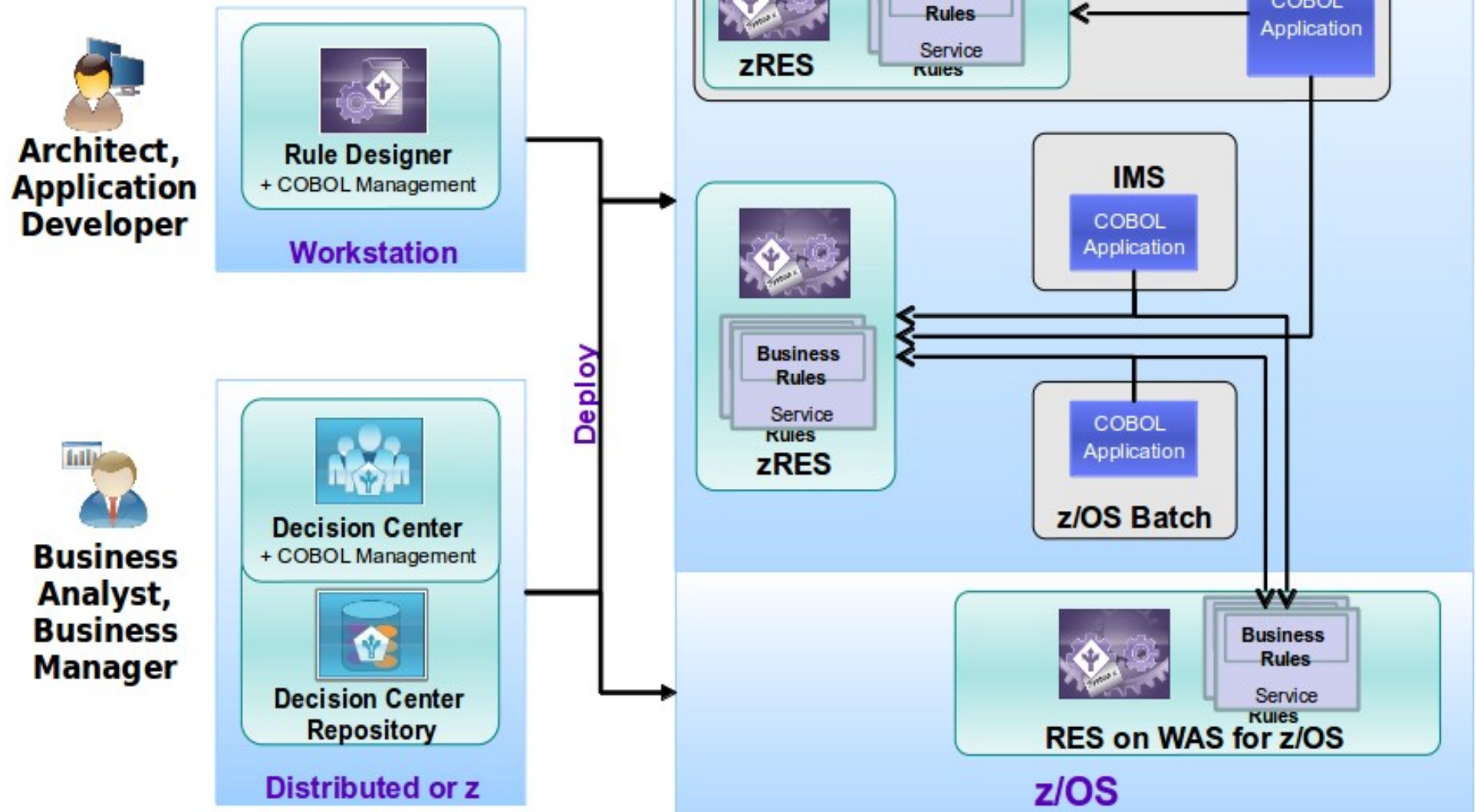


## The value of this solution for your z/OS applications

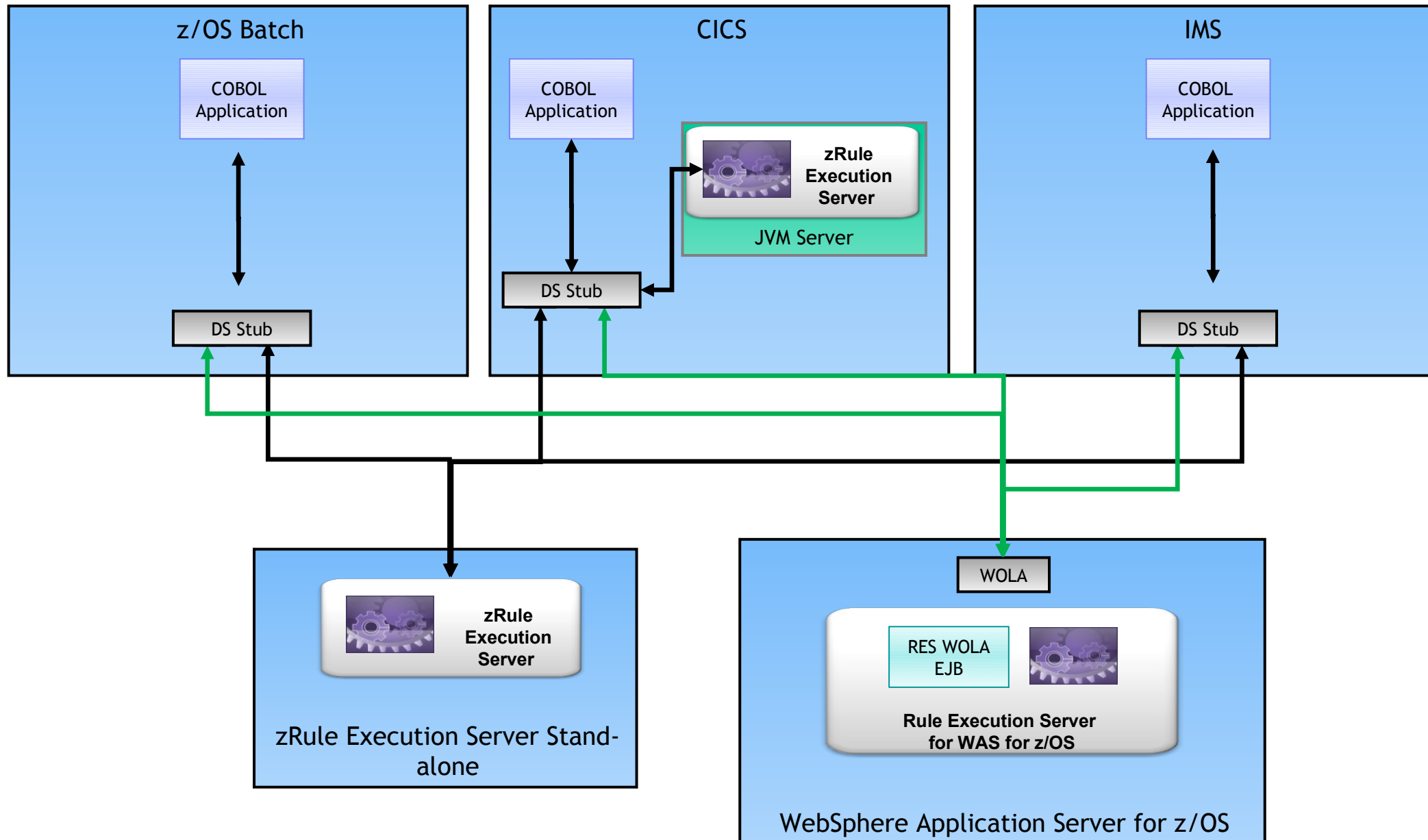
- *IBM Operational Decision Management enables organizations in every industry to make their business rules and business decisions clear, consistent and **expressed in business language** to be able to change when the business needs.*
- *Transformation or modernization of z/OS applications*
- *Ability to **react to change** (timely reaction to market and competitive changes)*
- *Overcome IT and Business mis-alignment – keep up and service **business requests***
- *Eliminate resource drain on application maintenance — **reuse of business decisions** across applications and platforms*



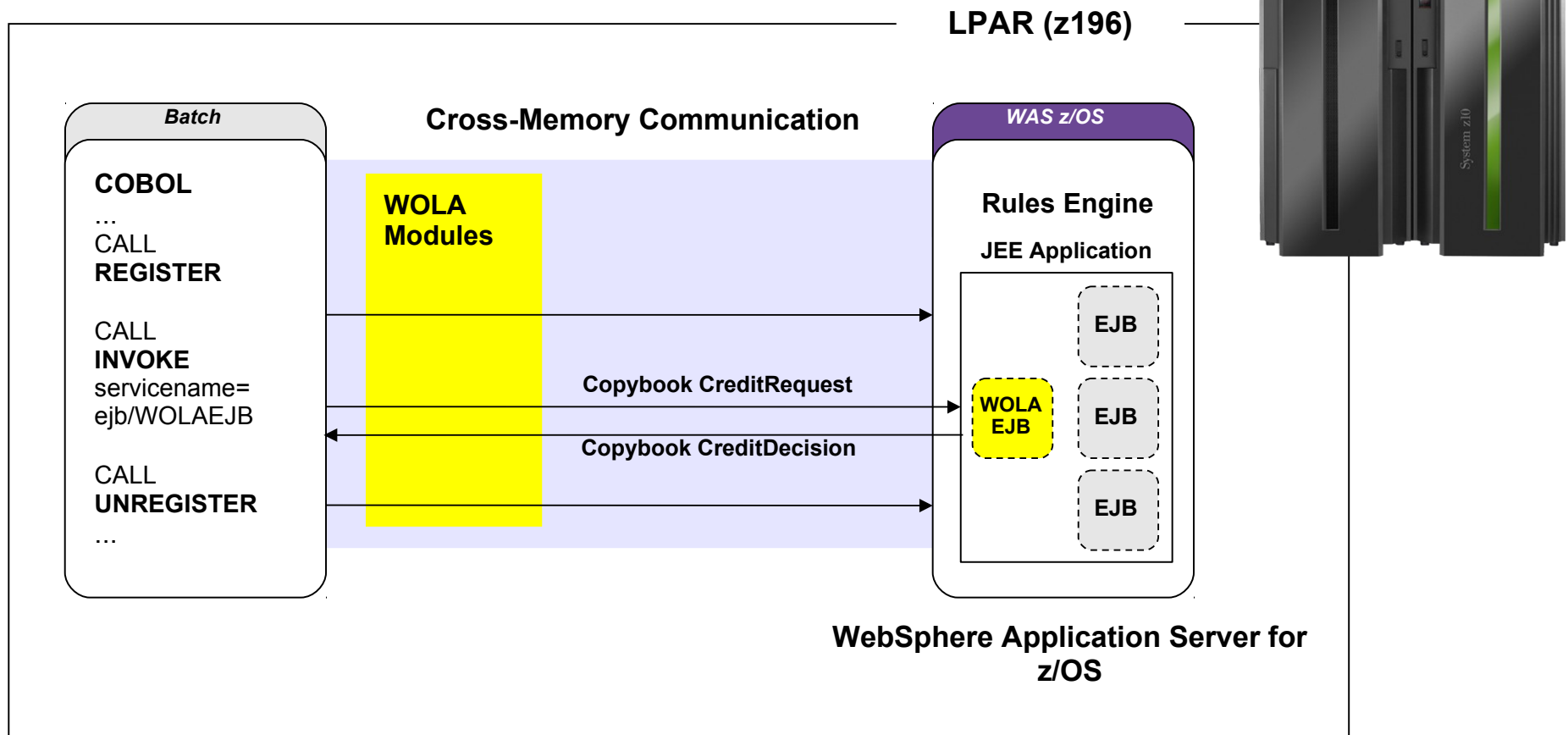
# Possible ODM Architectures for existing mainframe applications



# Rule Invocation Options for mainframe applications

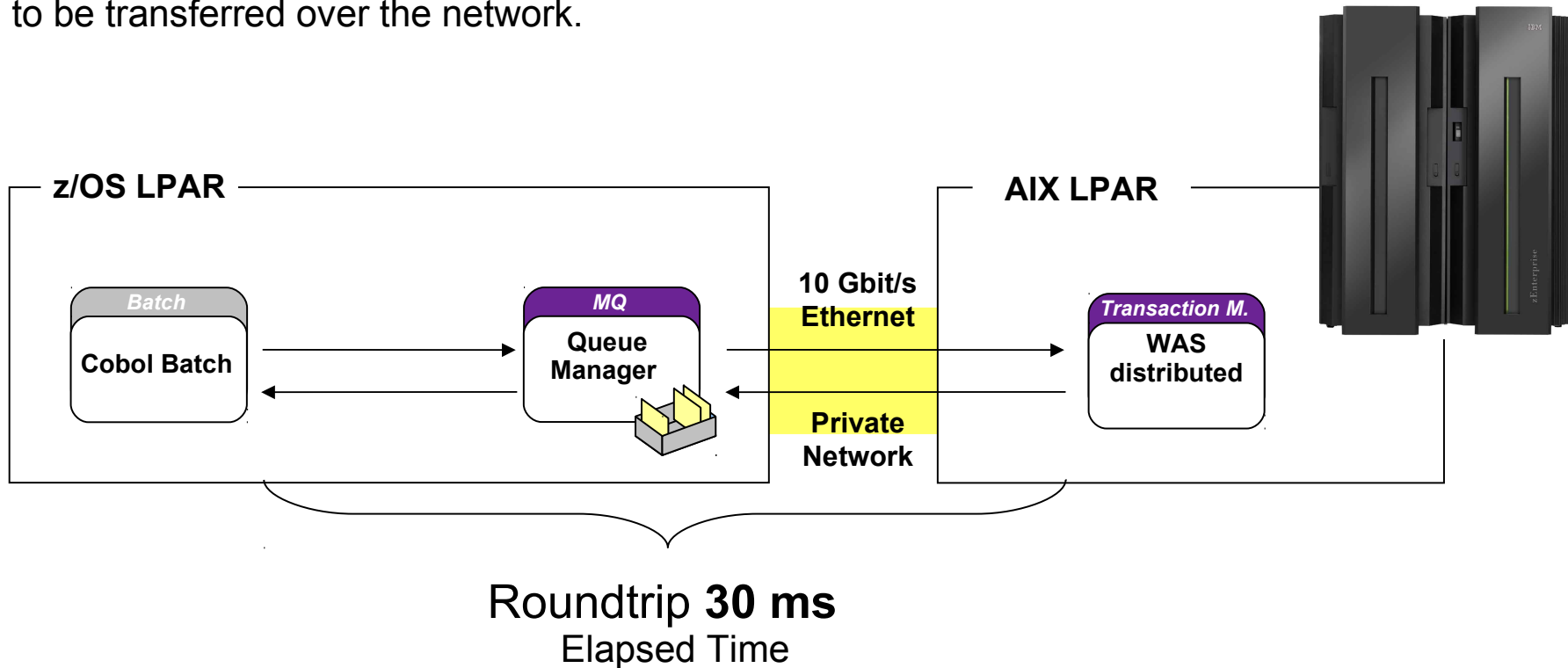


# Cobol Java Inter Language Communication With WOLA



## Szenario: Cobol Batch calls Rule Execution server on distributed

In this case the Cobol Batch Job calls business logic in a WebSphere Application Server distributed over MQ for each record. All requests need to be transferred over the network.

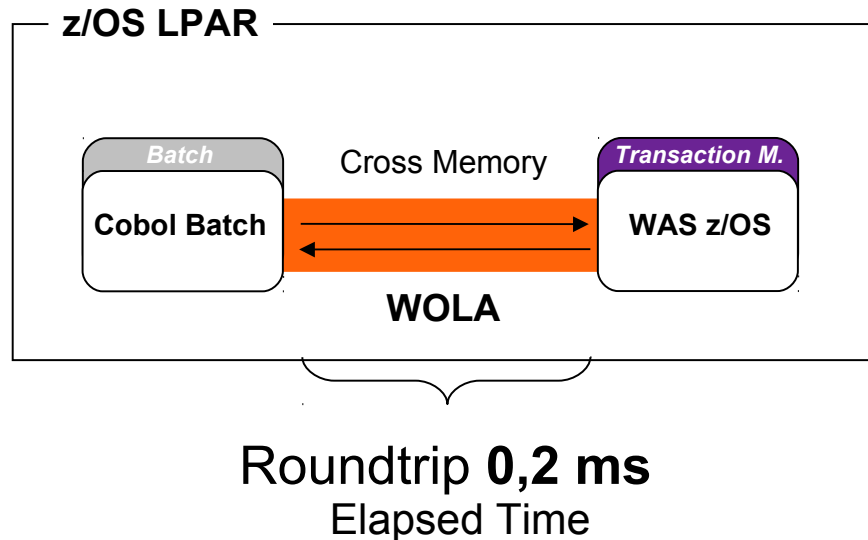


Job runtime caused by the infrastructure (without business logic) for 1.000.000 Batch records, which have to be processed.

$$1.000.000 \times 0,030 = 8,3 \text{ hours}$$

# Szenario: Cobol Batch calls Rule Execution server on z/OS

In this szenario the Cobol application and the WebSphere Application Server are located on the same LPAR and can communicate over a WOLA Cross-memory connection. This way of communicating is significant more efficient compared to TCP/IP and MQ, because it is a process to process communication using a shared memory.



**Especially for batch it is important, that all required data can be accessed local.**

If the batch job needs to call the distributed WebSphere Application Server for each record, the roundtrip elapsed time will be significant longer. This has an immediate impact on the job runtimes or transaction runtime.

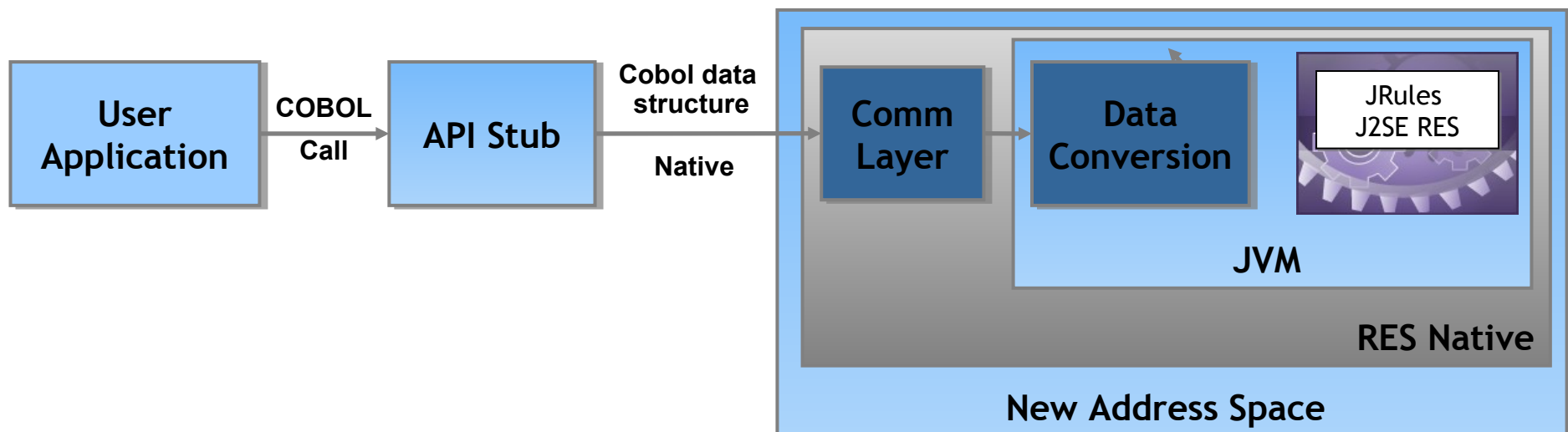
Job runtime caused by the infrastructure (without business logic) for 1.000.000 Batch records, which have to be processed.

$1.000.000 \times 0,000200 = 3,5 \text{ minutes (difference multiplier 150!!!)}$



# Overview of new zRule Execution Environment for z/OS

- Easy integration with existing COBOL applications running in CICS & Batch
- Designed for unique needs of the System z customer base
  - Native z/OS Execution in Java
  - COBOL stubs to glue the application to Rules Execution Native Server
  - Integrating core functionality from z based products
- Fully integrated with all key components of the existing BRMS offering
- For enhanced co-location, can implement within an existing CICS 4.x region



# Why modernize with ODMz and why now?

## *Benefits of the ODM Approach*

### *Modernization issues to resolve*

1. Consolidation of COBOL application portfolio
2. Be able to react to changes requested by business in days, not months.
3. Sharing rules across platform
4. Running parallel

### ✓ **Cost savings**

- ✓ More effective application development & maintenance with less business risk
- ✓ Consolidation/restructure of existing applications, saving hardware & resources
- ✓ Rule testing and simulation to ensure accuracy of changes prior to deployment which will minimize re-work

### ✓ **Changing ratio of source inventory to development skills**

- ✓ Forcing need for formal processes with an on line electronic repository

### ✓ **Improved agility**

- ✓ Decouple development and business rule lifecycles
- ✓ New rules to enforce new business policies to multiple applications

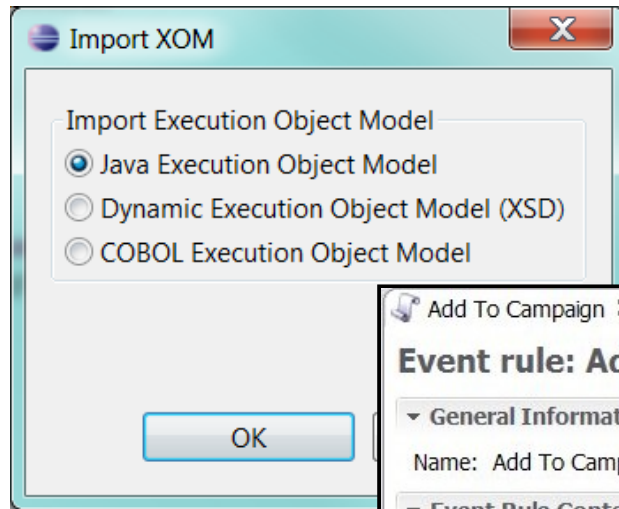
### ✓ **Incremental rule modernization: applying technology and process to gain increased “decision making” agility**

- ✓ Gradually pull out rules from existing applications - does not require a “big bang” change
- ✓ Rewrite business rules in natural language

## Rule & Event Designer

### ■ Eclipse-based Development Environment

- Rule Designer Perspective
- Event Designer Perspective
- Integrated support for COBOL

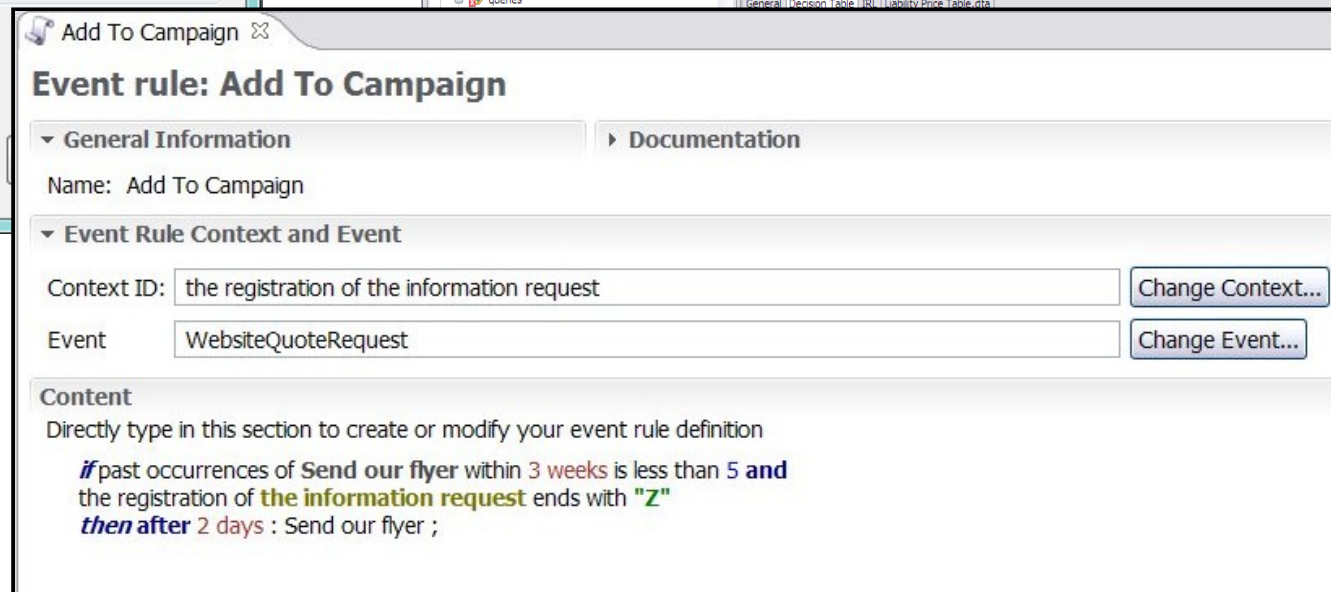


Rule - Pricing/rules/Pricing/Coverage Pricing/Base Premium/Liability/Liability Price Table.dta - Eclipse SDK

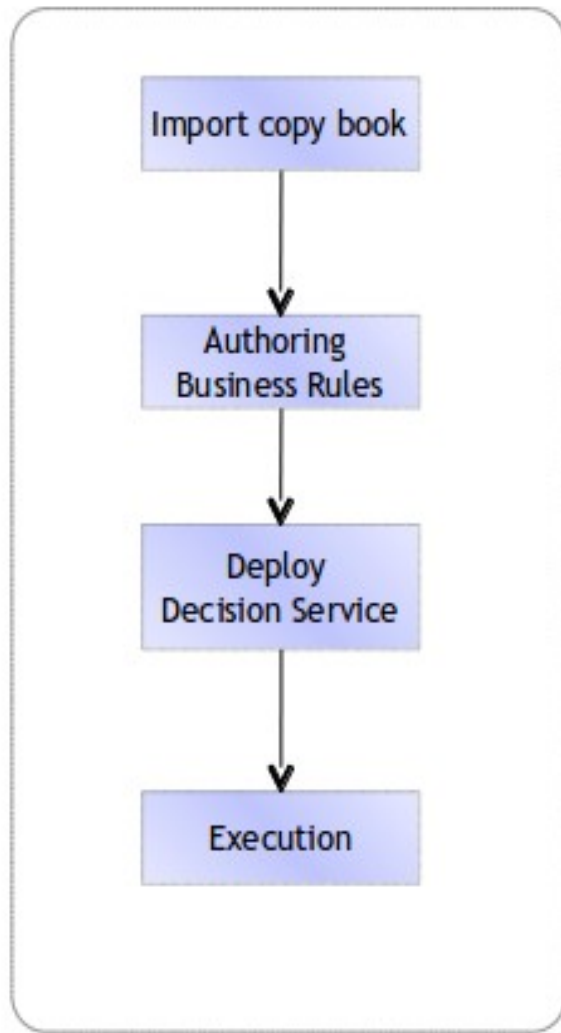
the limits of the coverage is min: \$25,000 and max: \$50,000

	Coverage Limit		Annual Mileage		Base Premium (\$)
	Min Limit	Max Limit	Min	Max	
0				< 5 000	\$100
1			[5 000	15 000[	\$105
2	\$15,000	\$30,000	[15 000	25 000[	\$110
3				≥ 25 000	\$125
4				< 5 000	\$110
5	\$25,000	\$50,000	[5 000	15 000[	\$115
6			[15 000	25 000[	\$120
7				≥ 25 000	\$135
8				5 000	\$115
9				15 000[	\$120
10				25 000[	\$125
11				25 000	\$140

**if**  
all of the following conditions are true :  
- the limits of 'the coverage' is min: \$25,000 and max: \$50,000  
- the annual mileage of 'the vehicle coverage request' is at least 25000  
**then**  
set base premium for 'the coverage quote' to \$135 ;



# Starting with a COBOL copy book



## Scenario

- Existing COBOL containing business rules
- Data model defined in COBOL copybook
- Use BRMS to modernize the business policy

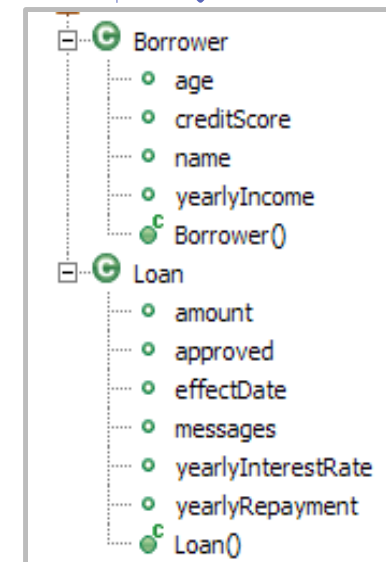
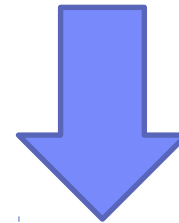
## Benefits

- Modernize business policies in BRMS
- Rules can be invoked 'naturally' from existing application
- Business policy/rule lifecycle detached from application lifecycle

## Rule Authoring – COBOL Copybook XOM

- Support Enterprise COBOL 3.4, 4.1 & 4.2
- A Java is created from the copybook structure
  - Java XOM & Java code to marshal between COBOL <-> Java
  - 01 level structures mapped to class in BOM
- Redefines statements supported
  - Select which redefines structure to import
- COBOL Table support
  - Mapped to Java **List<type>** structures
- COPY statements supported
- Level 88 supported
  - Mapped to methods in BOM

```
01 Borrower.  
   05 name                PIC X(20).  
   05 creditScore          PIC S9(10).  
   05 yearlyIncome         PIC 9(10).  
   05 age                  PIC 9(3).  
01 Loan.  
   05 amount               PIC 9(10).  
   05 yearlyInterestRate   PIC 99.  
   05 yearlyRepayment      PIC 9(10).  
   05 effectDate           PIC X(8).  
   05 approved             PIC X.  
   05 messageCount         PIC 9(2).  
   05 messages             PIC X(60)  
                           OCCURS 0 TO 99 TIMES  
                           DEPENDING ON messageCount.
```



## zRES: Business Rule Execution

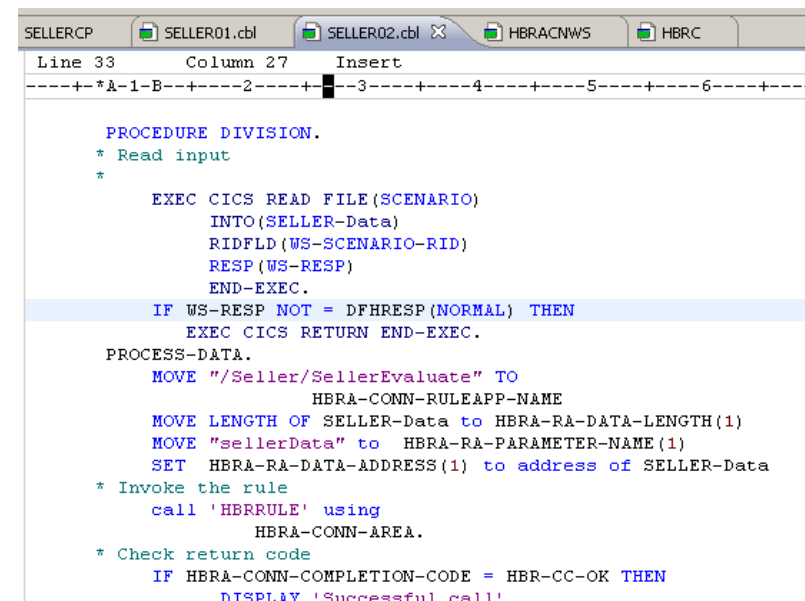
### Runtime enablement

- Write the Decision Service invocation in COBOL
- COBOL code remains independent of the Business Rules lifecycle on a stable decision service signature

### Decision Service Hot Deployment

- New decision version 'instantly' available
- From Rule Designer & Decision Center
- Versioned service made ready for execution from COBOL
- Let running executions complete

```
01 HBRA-CONN-AREA.
  10 HBRA-CONN-EYE          PIC X(4) VALUE 'HBRC'.
  10 HBRA-CONN-LENTH        PIC S9(8) COMP.
  10 HBRA-CONN-VERSION      PIC S9(8) COMP VALUE +1.
  10 HBRA-CONN-RESERVED01    PIC X(8).
  10 HBRA-CONN-FLAGS        PIC S9(8) COMP VALUE +1.
  10 HBRA-CONN-INSTANCE     PIC X(24).
  10 HBRA-CONN-RETURN-CODES.
    15 HBRA-CONN-COMPLETION-CODE PIC S9(8) COMP.
    15 HBRA-CONN-REASON-CODE     PIC S9(8) COMP.
  10 HBRA-CONN-RULEAPP-NAME  PIC X(256).
  10 HBRA-RA-PARMS OCCURS 32.
    15 HBRA-RA-PARAMETER-NAME PIC X(48).
    15 HBRA-RA-DATA-ADDRESS  USAGE POINTER.
    15 HBRA-RA-DATA-LENGTH   PIC 9(8) BINARY.
  10 HBRA-RESPONSE-AREA.
    15 HBRA-RESPONSE-MESSAGE PIC X(256).
  10 HBRA-RESERVED.
    15 HBRA-RESERVED02       PIC X(128).
```



```
SELLERCP  SELLER01.cbl  SELLER02.cbl  HBRA-CONN-AREA  HBRC
Line 33  Column 27  Insert
-----*A-1-B-+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----
PROCEDURE DIVISION.
* Read input
*
  EXEC CICS READ FILE(SCENARIO)
    INTO(SELLER-Data)
    RIDFLD(WS-SCENARIO-RID)
    RESP(WS-RESP)
  END-EXEC.
  IF WS-RESP NOT = DFHRESP(NORMAL) THEN
    EXEC CICS RETURN END-EXEC.
  PROCESS-DATA.
    MOVE "/Seller/SellerEvaluate" TO
      HBRA-CONN-RULEAPP-NAME
    MOVE LENGTH OF SELLER-Data TO HBRA-RA-DATA-LENGTH(1)
    MOVE "sellerData" TO HBRA-RA-PARAMETER-NAME(1)
    SET HBRA-RA-DATA-ADDRESS(1) TO ADDRESS OF SELLER-Data
  * Invoke the rule
    call 'HBRRULE' using
      HBRA-CONN-AREA.
  * Check return code
    IF HBRA-CONN-COMPLETION-CODE = HBR-CC-OK THEN
      DISPLAY 'Successful call'
```



## zRES: New Programming API

### \* Connect to Execution Region

```
call 'HBRCONN'  
    using HBRA-CONN-AREA
```

### \* Populate Header with parameter data

### \* Connect to Execution Server

```
call 'HBRRULE'  
    using HBRA-CONN-AREA  
IF HBRA-CONN-COMPLETION-CODE = HBR-CC-OK  
THEN  
    . . .
```

### \* Disconnect from Execution Region

```
call 'HBRDISC'  
    using HBRA-CONN-AREA
```

```
01 HBRA-CONN-AREA.  
10 HBRA-CONN-EYE          PIC X(4) VALUE 'HBRC'.  
10 HBRA-CONN-LENTH        PIC S9(8) COMP.  
10 HBRA-CONN-VERSION      PIC S9(8) COMP VALUE +2.  
10 HBRA-CONN-RETURN-CODES.  
15 HBRA-CONN-COMPLETION-CODE PIC S9(8) COMP.  
15 HBRA-CONN-REASON-CODE   PIC S9(8) COMP.  
10 HBRA-CONN-FLAGS        PIC S9(8) COMP VALUE +1.  
10 HBRA-CONN-INSTANCE     PIC X(24).  
10 HBRA-CONN-RULE-COUNT   PIC S9(8) COMP.  
10 HBRA-CONN-RULE-MAJOR-VERSION PIC S9(8) COMP.  
10 HBRA-CONN-RULE-MINOR-VERSION PIC S9(8) COMP.  
10 HBRA-CONN-RULEAPP-NAME PIC X(256).  
10 HBRA-RESPONSE-AREA.  
15 HBRA-RESPONSE-MESSAGE  PIC X(512).  
10 HBRA-RA-PARMETERS.  
15 HBRA-RA-PARMS OCCURS 32.  
20 HBRA-RA-PARAMETER-NAME PIC X(48).  
20 HBRA-RA-DATA-ADDRESS   USAGE POINTER.  
20 HBRA-RA-DATA-LENGTH   PIC 9(8) BINARY.  
10 HBRA-RESERVED.  
15 HBRA-RESERVED02        PIC X(12).  
15 HBRA-RESERVED03        PIC X(64).  
15 HBRA-RESERVED04        PIC X(64).  
15 HBRA-RESERVED05        PIC X(128).  
15 HBRA-RESERVED06        PIC X(128).
```

```

Line 81      Column 12      Insert      144 changes
-----+*A-1-8-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7-----
* Read scenario data
  MOVE ALL LOW-VALUES TO WS-IN
  UNSTRING SCENARIO-DATA DELIMITED BY ','
    INTO
      WS-IN-data(1) WS-IN-data(2) WS-IN-data(3)
      WS-IN-data(4) WS-IN-data(5) WS-IN-data(6)
* Populate the borrower from scenario data
  move WS-IN-data(1) to name
  Compute creditscore      = Function numval(WS-IN-data(2))
  Compute yearlyIncome     = Function numval(WS-IN-data(3))
* Populate the loan from scenario data
  Compute amount           = Function numval(WS-IN-data(4))
  Compute yearlyRepayment  = Function numval(WS-IN-data(5))
  Compute yearlyInterestRate = Function numval(WS-IN-data(6))

* Invoke the rule
  call 'HBRRULE' using HBRA-CONN-AREA

EXEC CICS SUSPEND END-EXEC

* Display rule responses, or error code, as appropriate
  if HBRA-CONN-COMPLETION-CODE = HBR-CC-OK then
    display 'HBR CALL Successful'

* Disconnect
  call 'HBRDISC' using HBRA-CONN-AREA

  IF HBRA-CONN-COMPLETION-CODE NOT EQUAL HBR-CC-OK THEN
    perform onFailedCall
  END-IF

  perform prtDemoText

EXEC CICS RETURN END-EXEC
GOBACK.

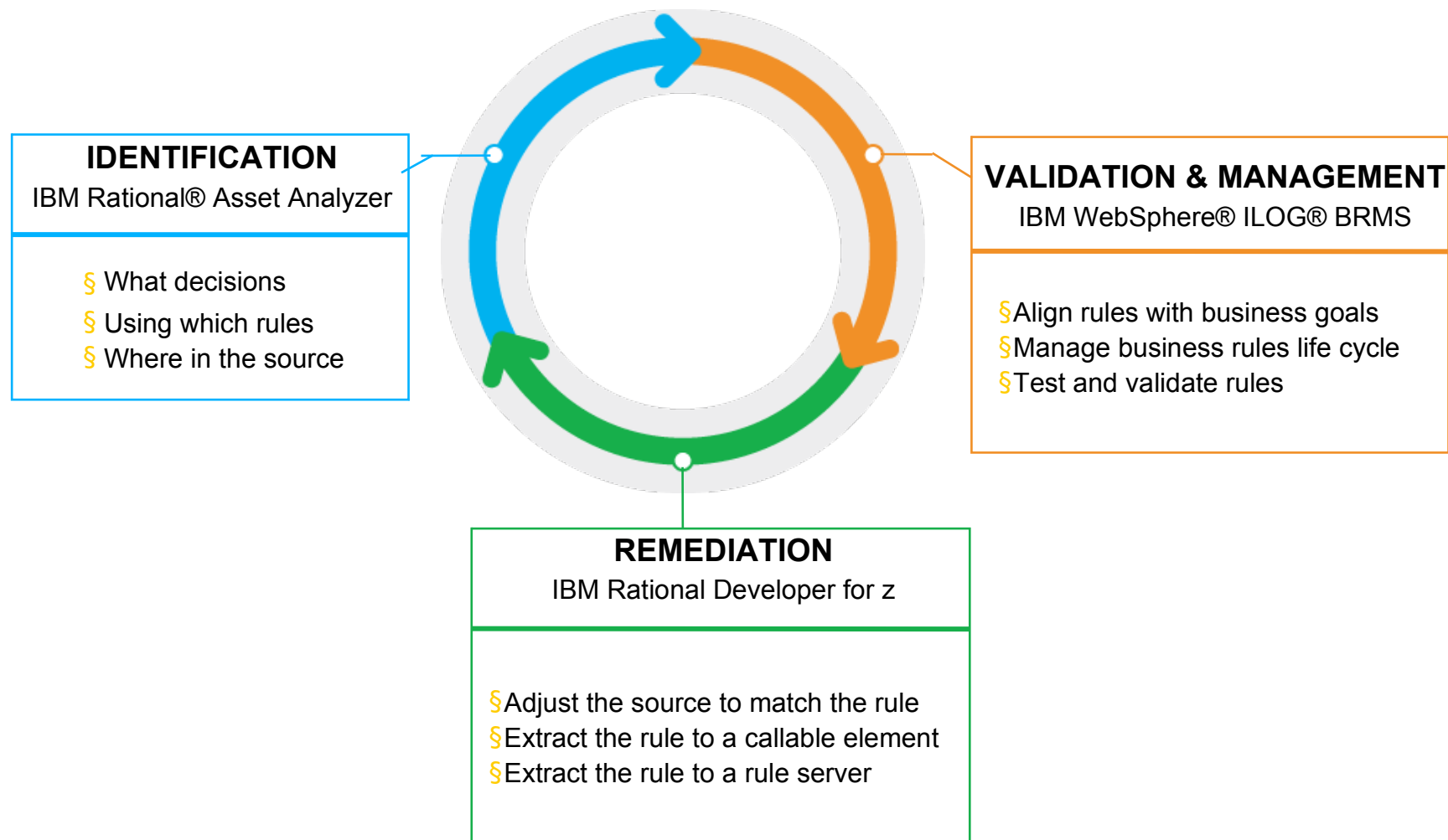
```

***Ich mag den Mainframe, weil wir ein gemeinsames Talent haben:  
Wir können viele verschiedene Dinge erledigen – gleichzeitig!***

*Isabel Arnold, Technical Sales IBM System z*



## Business Decision Modernization



### *Delivering...*

The essentials for business rule mining of existing software assets enhancing the ability to capture, maintain and take advantage of application knowledge that can provide insight into an application's structure and its interactions with business data.



## What is Rational Asset Analyzer?

An application understanding tool

§ Improved project effectiveness, with reduced risk and improved productivity

- Gather complexity metrics across multi-platform applications
- Determine the application structure and key relationships
- Identify scope and impact of pending application or database changes

### Role Oriented User Interface

- Developer-oriented Eclipse user interface integrated with RDz
- Easy-to-use browser interface for search, exploration, dashboard and construction of complex queries

### Comprehensive repository built on DB2

- Accessible via RESTful interfaces
- Data schema is documented
- Enabled for Rational Insight dashboard integration

### Fundamental to business decision making solutions

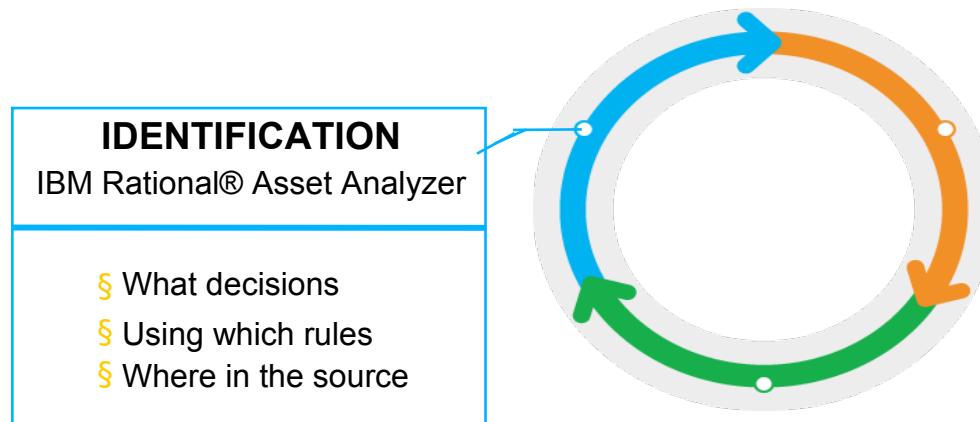
- Provides business rule identification in source code inventory
- Enables business rule capture and management with WebSphere ILOG BRMS

### Platform-specific editions available

- Rational Asset Analyzer - Windows server-based with z/OS access
- Rational Asset Analyzer for System z - z/OS server-based with Windows and AIX access



## Business Decision Mining with RAA V6



### — Scope the effort

- Defined Business Goal; with process model with specific set of business decisions
- Deliver incremental value: Keep a manageable size with timely deliverables

### — Establish the vocabulary

- Import Business Terms from ILOG BRMS or define them with RAA dialogs
- Map Business Terms to developer “terms”, code variables and data elements

### — Define the candidate business rules

- RAA will identify source statements that “act” on the business terms through their associated data elements
- Form unstructured candidate rules based on the identified source statements
- Create structured candidate business rules using ILOG Editor within RAA

### — Export Rules to ILOG



## Business Rule Modernization: Identification

```
31. 003100
32. 003200   IF 50
33. 003300     COMPUTE C
34. 003400   ELSE
35. 003500     DISPLAY '
36. 003600
37. 003700   GOBACK.
38. 003800/
39. 003900 100-FACTOR2.
40. 004000
41. 004100   IF C-AGE > 5
42. 004200     COMP
43. 004300   ELSE
44. 004400     SET
```

### Relate to a Business Rule

Relate the IF Statement to a business rule.

Business rule:

Provide discount for seniors

The statement has the following data elements with related terms and term properties:

### Rational Asset Analyzer

Home Explore Impact analysis Database

Context : Explore rule mining assets Business rule summary Business rule details

#### Business rule details

Actions Select an action

**Details**

Name: Provide discount for seniors

Documentation: If a person's age is over some threshold, give them a discount.

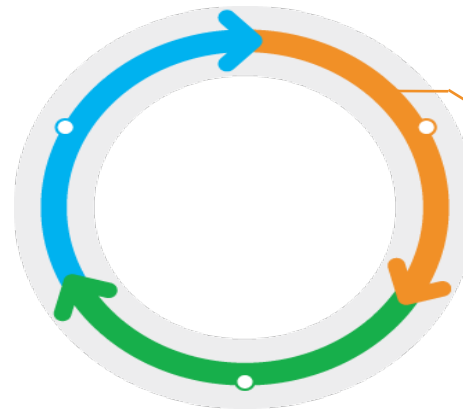
**Related statements (1)**

Statement	Program	Relationship type	Source location	Site
IF	DISCOWE	User asserted	C:/brm/testdata/BusinessRuleMining/src/DISCOWE.CBL line 41	HYSITE

**User-related assets (0)**

Aug 10, 2010 5:37:20 PM

## Business decision Mining with RAA V6



### VALIDATION & MANAGEMENT

IBM WebSphere® ILOG® BRMS

- § Align rules with business goals
- § Manage business rules life cycle
- § Test and validate rules

### — Validate the Rules

- Review candidate rules with business analysts
- Establish what the rule “should be” vs “what it is” in the source
- Build rule project in Rule Studio
- Within ILOG BRMS, capture any rule project revisions

### — Analyze and test rules

- Using rule analysis in Rule Studio reconcile any conflicting rules
- Consider value/impact of sharing rules
- Test and Simulate rule project in ILOG BRMS to validate business outcome

### — Select Source Code Remediation Options

- Update application source to conform to ILOG BRMS implemented rules
- Update application source to call ILOG Rules for COBOL generated program
- Update application source to call ILOG Rule Execution server, directly or via web service